Computer Science Department
cs.salemstate.edu

**SYLLABUS** **Fall, 2019**

**CSC 260 Data Structures and Algorithms** **4 cr.**
**Prerequisites: CSC101 or CSC 200A, and CSC115 or CSC 202J**

**Instructor**: Beifang Yi **Office**: MH 211A **Phone**: (978) 542-7246
**email**: byi@salemstate.edu **Hours**: TR (12:10pm-1:40pm) **Web Site**: http://cs.salemstate.edu/~byi/
W (1:40pm-3:40pm)

| Section | Time | Room | Final Exam |
|---|---|---|---|
| **01** | W & F 10:50am-12:05 pm | MH 206 | **Dec 16, Monday, 11:00am-1:00pm MH 206** |
| **L21** | W & F 12:15-1:30 pm | MH 201 | |

**Catalog description:**
Basic data structures such as stacks, queues, linked lists, and trees are studied and applied to problems in data storage and manipulation. Applications include basic searching and sorting algorithms. Fundamental strategies for algorithm design are reviewed and extended. Design, analysis and implementation techniques are discussed. Three lecture hours and three hours of scheduled laboratory per week, plus extensive programming work outside of class.
**Prerequisites:** CSC101 or CSC 200A, and CSC115 or CSC 202J.

**Goals:**
The purpose of this course is to develop students' knowledge and appreciation of organization and retrieval techniques and to familiarize students with the basic concepts of order-of-magnitude analysis. The goals of this course are:

CG01: to develop an appreciation for the process of data abstraction and its usefulness in software development;
CG02: to develop the skills and knowledge necessary to perform design and basic analysis of algorithms;
CG03: to present a selection of the most common data structures and their standard implementations and uses;
CG04: to present a selection of the most common algorithms for searching and sorting.

**Objectives:**
Upon successful completion of the course, student will have:

CO01: applied data abstraction techniques;
CO02: implemented several classic data structures "from scratch";
CO03: demonstrated knowledge and use of ADTs available in one or more language libraries;
CO04: recognized the factors required to perform algorithm design, analysis of algorithms and performed order-of-magnitude analysis;
CO05: chosen, with justification, an appropriate structure to match the requirements of a given problem, implemented the structure if necessary, and used it in an appropriate way to solve the problem;
CO06: utilized standard techniques for program validation;
CO07: demonstrated the ability to use the UML modeling language;
CO08: produced documentation for at least one major completed project, including formal class diagrams and rigorous test set specification and results;
CO09: participated in at least one group project involving problem analysis and design specification and selection;
CO10: demonstrated recognition of the need for future professional development through research into future trends in the areas of analysis of algorithms and application development and profiling.

**Course Topics:**

A detailed topics list and a general course bibliography can be found on the Computer Science Department website at http://cs.salemstate.edu/dept/index.php?page=184. Select CSC 260 to access a PDF document.

**Text:**

**(Required) Data Structures and Algorithms in Java**, 6nd Edition, by Michael Goodrich, Roberto Tamassia, and Michael Goldwasser, Wiley, 2014 (ISBN: 978-1-118-77133-4).

**Required Material:**

**(Required)** Thumb (flash) drive, 16 GB minimum or online storage (for saving your projects)

**Additional references:**

- Course website: http://cs.salemstate.edu/~byi/CSC260/.
- **Java Software Structures: Designing and Using Data Structures**, 4th Edition, by Lewis and Chase. Pearson, 2014
- **Data Structures and Algorithms in Java**, 2nd Edition, by Robert Lafore. SAMS, 2003
- **Data Structures and the Java Collections Framework**, 3rd Edition, by William J. Collins. Wiley, 2011

**Software:**

(**Required**) J2SE 8.0 (or above) and NetBeans 8.0 (or above) (this is the only IDE that is installed on the lab machines; you may use your selected IDE). Free copies of the software that have been customized for the course can be downloaded in the Department labs. There are many IDEs for Java programming and you have the option of choosing anyone or none.

**Cell phones:**

Turn the ringer off, or, better yet, *turn the phone off*.

**Lecture/Lab Attendance:**

Class policy is that of the Registrar's office- see the University catalog for details. **Note** that *you are at all times responsible for materials and assignments discussed in class*. We will use SSU's online course management system, Canvas (https://elearning.salemstate.edu/) to post assignments, test grades, and announcements regarding the course topics and progress. You will need to visit Canvas (with your SSU Navigator use-name and password) for the course activities. Canvas uses your ***SSU-stored email*** for the communication between you and the instructor and thus you ***must use this email address***. Each student is responsible for completing all course requirements and for keeping up with all that goes on in the course (whether or not the student is present).

**Class attendance** is ***strongly recommended***. Some of the class hours will be used to review important course topics, to discuss and investigate Java implementation details that time may not permit to be fully presented in the texts (for design and implementation drills, for programming exercises) to assist with design and implementation problems that arise in project exercises, to discuss and review important topics/questions regarding the tests, and to check/examine/grade the exercises and homework.

**Student-Instructor Communication:**

Data Structures and Algorithm Analysis is one of the fundamental subjects in computer science and also one of the core requirements in software design and development. Learning how to develop software is very much a **hands-on, experiential process** - the only way to be sure that you understand the material is to apply it by designing and writing programs.

If you have any questions regarding course material, and *in particular if you are having problems with a programming assignment*, the most effective way to get assistance is to *discuss with the instructor* (*either in the class, in the lab, or outside the classroom*).

Please **note**: Canvas is used for submission of the assignments/projects and for posting grades. If you ask questions through Canvas-grading-submission features, they usually do ***not*** go to the instructor. Please ask in the class/lab or send emails to the instructor!

**Final Grade:**

Final grade will be determined using the following grading weights:

| | |
|---|---|
| assignments (reading, writing, lab-exercises) | 25% |
| programming projects | 25% |
| Mini-tests | 25% |
| final examination | 25% |

**Please note**: the instructor will take attendance even though attendance is not used to calculate the final grade; the attendance will

be reported to the college as a required portion of your overall academic evaluation; however; you are at all times responsible for assignments and materials presented in class.

The following table shows how the course work is assessed against the course objectives:

| | Quizzes/Exams | Assignments (labs, programming projects, and other exercises) |
|---|---|---|
| CO01 | ✓ | ✓ |
| CO02 | ✓ | ✓ |
| CO03 | ✓ | ✓ |
| CO04 | ✓ | ✓ |
| CO05 | ✓ | ✓ |
| CO06 | | ✓ |
| CO07 | ✓ | ✓ |
| CO08 | ✓ | ✓ |
| CO09 | | ✓ |
| CO10 | | ✓ |

**Assignments (reading, writing, lab-exercises):**

There will be weekly various types of assignments: reading, writing, and lab exercises.

These assignments are designed to help understand the course topics, prepare for programming practices, and get prepared for the tests (i.e., the mini-tests and final examination). And **more important**: many of the test questions will be much like these assignment questions!

Readings will be assigned from the text on a regular basis: for the maximum benefit from reading, do the readings before the material is covered in class, and you will find that most of the assignment questions are based on the reading of the text.

There will be short-answer questions (writing assignments) which are designed to test your understanding of the course topics and which will prepare for your examinations.

Lab exercises are designed not only for help on the understanding of the course topics but also for the preparation of the programming projects. Please note that lab exercises and programming projects are different assignments from the programming projects and credits earned from the programming projects do not count for your lab assignment grades.

**Programming Projects:**

Programming projects will be assigned during the semester. *Most of them will have pre-lab activities to be completed prior to the implementation of the assigned tasks*. They *will definitely require* ***significant*** *programming time outside of scheduled lab*. Programming assignments may have different full score points, depending on the difficulty and the amount of the work of the exercises.

Each programming project must be submitted at Canvas (https://elearning.salemstate.edu/ ) and **must be tested/examined /checked in the lab by the instructor for full credits** of the project.

**Please note:** to get *full credits* for each project, **you must (1) submit your solution (that is program source code for programming projects) to Canvas by the deadline, (2) show your solution in the lab, (3) have the instructor test your code in the lab, (4) answer the questions provided by the instructor correctly during the testing, (5) modify/update your code correctly as a solution for a programming question that is very similar to the one you have just submitted/shown, (6) and sign your names/initials on the project-grading sheet** provided by the instructor. Should something unexpected or emergent happened that prevents you from showing your solutions and signing your names/initials on the grading-sheet on time, this is what is called "documented/emergency situation" –please refer to Due Dates/Times below. Also note that the instructor will post the assignment or project grades regularly (usually within one week after the assignment/project's due date) on Canvas and you are encouraged to check your assignment/project grades regularly.

- Given the lengthy steps of testing your solution code in the lab, please do submit your answers by deadline and be sure that

you are very familiar with your solution—the Java code of *your own work*!

- If during the test in the lab you cannot answer the provided questions correctly or make necessary changes upon your submission for a varied problem (which is very similar to the project problem), your scores for that project will be severely affected (which can be as low as **0%**).
- Testing and grading will be done *after* the deadline. If the testing is done and the grading is posted and you would improve on your grades by submitting new work, the updated grades will be based on the new submission time (i.e., subject to late-submission penalty). So please start working on the projects early and have the instructor test your solutions well before the deadline and you may have chance of improving on the projects (based on the test feedback) and also on the grades.

**Tests (mini-tests and final examination)**:

There will be several mini-tests administrated during the semester and one *comprehensive* final examination. Please refer to Final Grade above for the grading weights of these tests.

The mini-tests will be like closed-book written quizzes and small programming projects/exercises which must be completed during the lab hours within limited time amount (usually, less than 55 minutes) in the lab. These programming mini-tests will be much like lab exercises and/or programming projects on a smaller scale.

**Missed Tests:**

Tests may not be made up except for *documented/emergency* situations. If a test must be made up, arrangements must be made with the instructor to take the test before it is discussed in class (usually within a week of the test being administered).

**Due Dates/Time:**

- Late submission of assessed coursework will result in penalties defined in the following:
  - **For every day the assignment is late after due date/time, 25% of the maximum will be deducted from the assignment score. Assignments that are more than three days late will be given no credit.**
  - A grade of **zero** will be given for any assignments including projects which are submitted after the final examination time. That is: **no assignments / projects (and other types of coursework) will be accepted after the final examination**.
- Should there be an emergency that prevents you from completing/submitting your assignments/projects on time, you will need to send *an email request* for the extension on the coursework submission. The instructor will reply to this request email with a specific number of days for the extension period or a new deadline for the assignment and you will need to keep *this email as a record of the extension approval*. Only a request email does ***not*** guarantee the extension approval

**Study Groups:**

While I strongly encourage study groups, I require that each student hand in his/her answers in her/his own words - if two answers are highly similar to each other, neither will receive credit.

When working on your programming projects, you may discuss with others the project topics, the algorithms and methodologies related to the project; but when you work on writing the code, this coding work should be 100% of your own work. **If two answers/written code segments come out exactly the same or highly similar, neither will receive credit and/or further actions will be taken** (such as reporting to the department and/or university).

**Academic Integrity:**

Academic Integrity Policy and Regulations can be found in the University Catalog and on the University's website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy* and will be dealt with according to the University's formal Procedures.

---

All students are expected to be familiar with the academic regulations, including those regarding Academic Integrity, for Salem State University as published in the college catalog. In addition, each student is responsible for completing all course requirements and for keeping up with all that goes on in the course (whether or not the student is present).

Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately. Students with Disabilities who have not previously done so should provide

documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services.

In the event of a University declared critical emergency, Salem State University reserves the right to alter this course plan. Students should refer to salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a university declared critical emergency. In the event of an emergency, please refer to the alternative educational plans for this course located at Canvas (https://elearning.salemstate.edu/). Students should review the plans and gather all required materials before an emergency is declared.

Please remember that if, for any reason, you decide to drop this course, you **MUST** do so officially through the Registrar's office. The last day to withdraw from a course this semester is **November 22nd**.

**Note:** This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via all regular class communication mechanisms (class discussion, emails, and/or the course link at Canvas https://elearning.salemstate.edu).